Security of AI along the entire development cycle

Bastien Vuillod, Assaad Al Saneh, Kevin Hector, Pierre-Alain Moëllic, Cedric Gouy-Pailler

CEA LETI, Mines Saint-Etienne, CEA LIST

ML security along the entire cycle

The model you deploy is not the model you evaluated.

Between training and inference, a model is fine-tuned, aligned, compressed, copied into memory and executed on physical hardware. The attack surface is increasing at each step.

1 Pre-training & supply chain

public checkpoints, model hubs

⚠ poisoned / trojaned checkpoints

2 Alignment & adaptation

SFT, RLHF/DPO, LoRA, federated fine-tuning

⚠ backdoors, safety removal

3 Compression

quantization, pruning, distillation

⚠ behaviour triggered by compression

4 Deployment

weights in DRAM / Flash on GPU, MCU

⚠ fault injection: Rowhammer, laser

5 Inference

API or physical access

⚠ adversarial examples, model extraction, side channels

Integrity	backdoors, misalignment, accuracy collapse
Confidentiality	model / parameter extraction, data leakage
Availability	denial of service, resource exhaustion

Why the embedded world makes it harder

- Physical access: the attacker can fault or probe the device
- 8-bit / 4-bit models: each bit carries more meaning
- AI Act, Art. 15: high-risk systems must resist poisoning, adversarial inputs and model flaws, so evaluation and certification methods are needed

High-risk AI, AI Act Article 15

- High-risk AI systems shall be designed and developed in such a way that they achieve an appropriate level of accuracy, robustness, and cybersecurity, and that they perform consistently in those respects throughout their lifecycle.

  […]

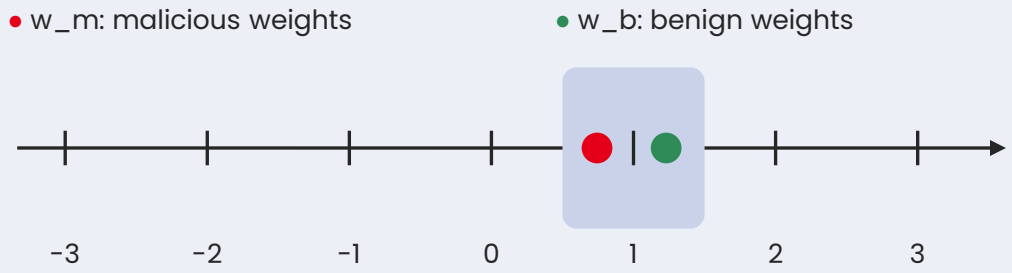
- High-risk AI systems shall be resilient against attempts by unauthorised third parties to alter their use, outputs or performance by exploiting system vulnerabilities. The technical solutions aiming to ensure the cybersecurity of high-risk AI systems shall be appropriate to the relevant circumstances and the risks. The technical solutions to address AI specific vulnerabilities shall include, where appropriate, measures to prevent, detect, respond to, resolve and control for attacks trying to manipulate the training data set (data poisoning), or pre-trained components used in training (model poisoning), inputs designed to cause the AI model to make a mistake (adversarial examples or model evasion), confidentiality attacks or model flaws.

Positioning: HW/SW interface

	Pre-training / supply chain	Alignment & adaptation	Compression	Deployment	Inference
Algorithmic (software)	*Public pre-trained model = attacker's offline knowledge*	③ **Security through alignment stages** safety alignment erased by fine-tuning, quantization *A. Al Saneh*			
HW/SW interface		④ **Federated adaptation** LoRA & backdoor lifespan → bit-flip poisoning of one client (Chain-of-Bit-Flips) *B. Vuillod · K. Hector · P-A. Moëllic*	② **Quantization-based attacks** benign in FP, malicious once quantized *A. Al Saneh*	① **Budget-aware bit-flip attacks** Rowhammer (DRAM), laser (MCU Flash): integrity & confidentiality; how many faults does the attacker really need? *K. Hector · P-A. Moëllic*	
Physical (hardware)			*Number format (INT8, NF4, GGUF) fixes which bits exist, and how much each one matters*		

Quantization and hardware faults

Quantization is a many-to-one map



every full-precision value that rounds to q = 1. **Moving inside it changes the FP model but not the quantized one.**

- Block-wise scales (32–256 weights) in NF4, FP4, GGUF k-quants
- Users quantize locally

Fault models: which bits can the attacker change?

Rowhammer	DRAM; bit-flip; remote/software-induced, sparse vulnerable cells
Laser	MCU Flash / SRAM; bit-set; physical access, precise in space and time
Glitch / EM	clock, voltage, EM pulses; instruction skip, data corruption

Attacker budget = number of faults × spatial / temporal constraints of the injection means.

Bit-flip attacks and attacker budget

Hardware faults on embedded DNNs: K. Hector PhD (2024)

Concept · Bit-Flip Attack

Gradient-guided progressive search of the most damaging bits of an 8-bit quantized network (Rakin et al., ICCV'19).

13 bit-flips

out of 93 M bits drive ResNet-18 on ImageNet from 69.8 % to 0.1 % top-1.

But: no budget in the threat model, and the goal (random guess) ignores what a laser or Rowhammer can actually do.

Evaluation pitfalls

IOLTS'22

BFA results vary a lot, and learning rate and architecture (MLP vs CNN) change its behaviour, giving a false sense of robustness. Weight clipping and RandBET are cheap defenses that work.

From simulation to laser

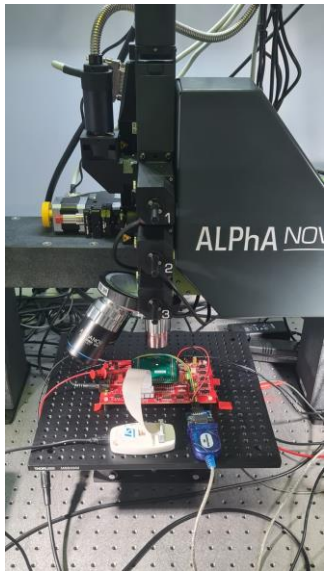
SAFECOMP'23

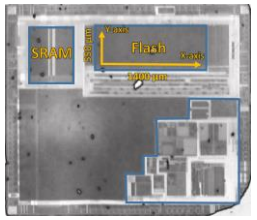
First laser fault-injection evaluation of parameter-based attacks on a 32-bit Cortex-M (weights in Flash): bit-set fault model and a BFA adapted to the memory layout (one shot cannot reach every bit-line).

Faults against confidentiality

ESORICS-W'23

Laser + Safe-Error Attack recovers the most significant bits of 8-bit weights; a substitute model trained with < 10 % of the data mimics the victim, and adversarial examples transfer almost perfectly.





Left: laser bench. Above: IR view of the Cortex-M die (SRAM, Flash)

≥ 80 % of the 6 MSBs extracted

86–96 % fidelity of the substitute model

Hector, Moëllic, Dumont, Dutertre, IOLTS'22

Dumont, Hector, Moëllic, Dutertre, Pontié, SAFECOMP'23

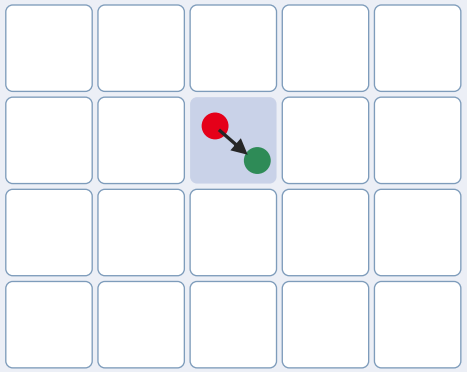
Hector, Moëllic, Dutertre, Dumont, ESORICS'23 Workshops

Hector, PhD thesis, Mines Saint-Étienne, 2024

Quantization to activate hidden behaviour

Compression as an attack trigger

The 3-step attack (Egashira et al., NeurIPS'24)



1. **Inject:** fine-tune a malicious model θ_m
2. **Constrain:** keep its quantization cell: {θ : Q(θ) = Q(θ_m)}
3. **Repair:** PGD towards benign behaviour, projected in the cell to θ_b

FP32 θ_b: benign, passes the evaluation on the model hub

Q(θ_b) = Q(θ_m): malicious as soon as the user quantizes

Scenarios: insecure code, content injection, over-refusal, jailbreak

ELQ · NeurIPS'24

INT8 · NF4 · FP4

zero-shot rounding schemes are exploitable

Mind the Gap · ICML'25

88.7 %

Δ insecure code on GGUF (llama.cpp, ollama)

ACL · 2026

97.7 %

jailbreak ASR with a contrastive objective

Newer attacks

- Outlier injection makes the gap wider and needs no cell constraint (Widening the Gap, 2026)

Defenses: change the rounding

- EFRAP (Li et al., CVPR'24), QuantGuard (CCS'26): flip well-chosen rounding decisions; Q-resafe (Chen et al., ICML'25): safety patching

Ongoing · A. Al Saneh PhD: how robust are quantization-conditioned backdoors, and how should LLM safety be measured?

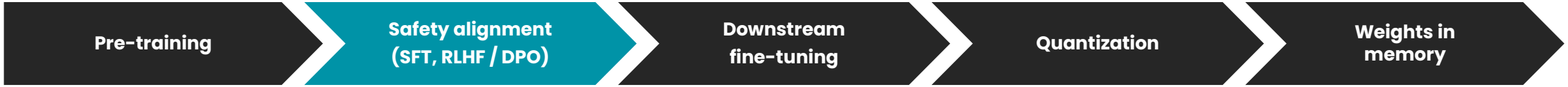
Egashira, Vero, Staab, He, Vechev, NeurIPS'24 & ICML'25

Song et al., ACL, arXiv 2601.02680

Widening the Gap, arXiv 2605.15152

Risks on alignment

Security through alignment stages



Safety is a property of **one checkpoint**, not of the model family.

Alignment is shallow

it mostly shapes the first few output tokens

Qi et al., ICLR'25

10 examples

fine-tuning jailbreaks GPT-3.5 Turbo; even benign fine-tuning erodes safety

Qi et al., ICLR'24

Q-misalign

misalignment dormant in FP, activated by quantization, survives later fine-tuning

Dong et al., ICLR'25

5–25 bit-flips

jailbreak aligned LLMs at runtime; 69–91 % ASR with Rowhammer on a GDDR6 GPU

PrisonBreak, Coalson et al., 2024

Our perspective: attacker budget × alignment

Quantized LLMs have far fewer bits to search so faults become realistic targets for integrity (jailbreak, misalignment) and availability (unbounded consumption).

3 bit-flips

LLaMA3-8B (8-bit): MMLU 67.3 % → 0 %; 28 flips for a 4-bit model

AttentionBreaker, Das et al., TMLR'25

Qi et al., Fine-tuning aligned LMs compromises safety, ICLR'24

Qi et al., Safety alignment should be made more than just a few tokens deep, ICLR'25

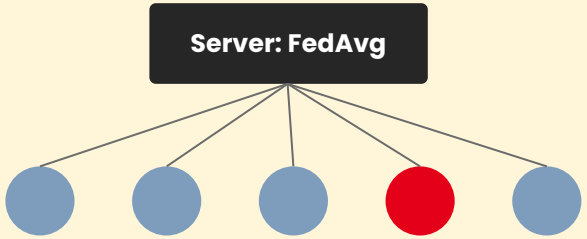
Dong et al., Durable quantization-conditioned misalignment, ICLR'25

Coalson et al., PrisonBreak, arXiv 2412.07192

FL adaptation: LoRA rank and backdoor lifespan

Federated model adaptation · B. Vuillod PhD

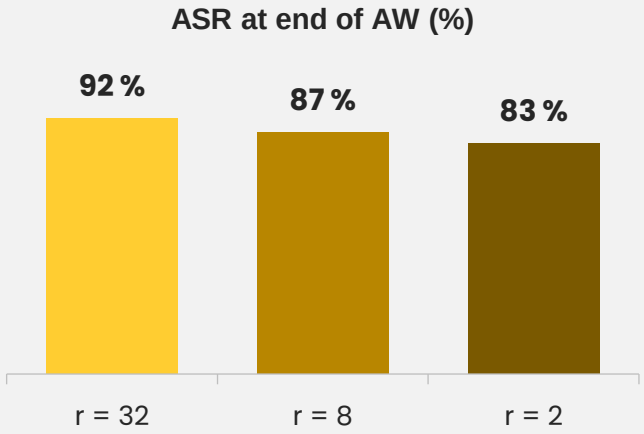
Basics



a malicious update is diluted among |P| clients

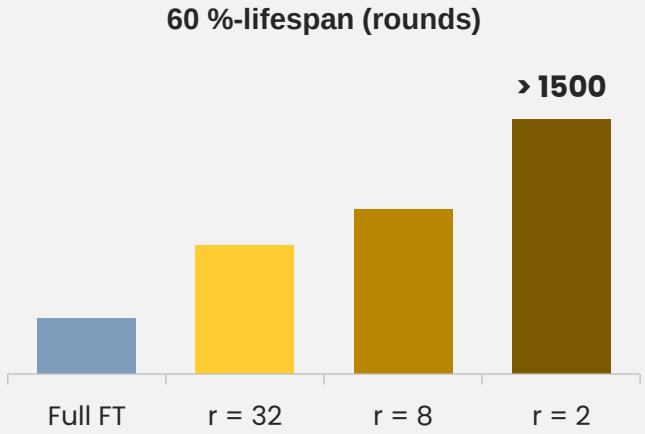
- **LoRA:** $\Delta W = B \cdot A$ with rank $r \ll d$; only A, B are trained ($r = 2$: 81k parameters)
- **Attack window (AW):** rounds in which malicious clients participate
- **Lifespan:** rounds after the AW during which the ASR stays above 60 %

Naive: short window AW = [0, 30]



Lower rank looks **safer**…

Converged: AW = [0, 200]



…but the backdoor **lives longer** with lower rank

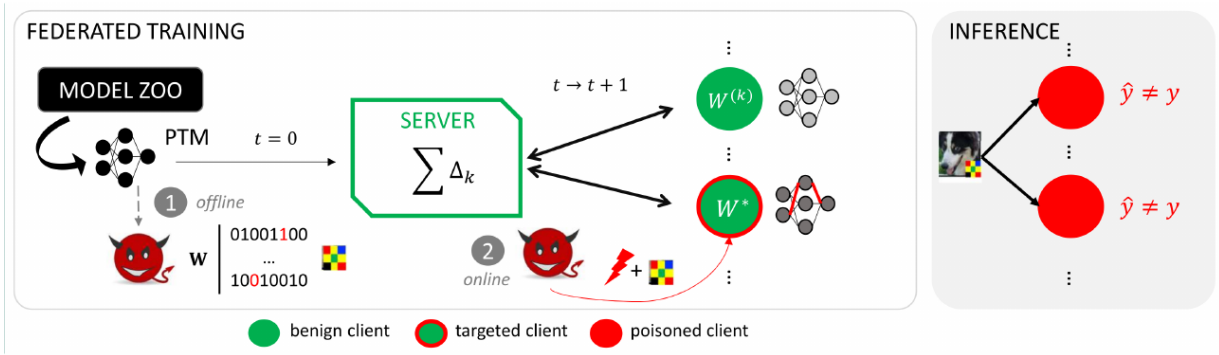
low rank slows down both backdoor injection and its erasure. Lifespan must be measured after the backdoor has converged, otherwise the conclusion is reversed.

ViT-B/16, EuroSAT (similar on CIFAR-10, GTSRB)
100 clients, 10 per round, 1500 rounds
4 attacks: baseline, DBA, A3FL, Neurotoxin (baseline shown)

Vuillod, Moëllic, Dutertre, Watch out for the lifespan: evaluating backdoor attacks against federated model adaptation, FPS 2025 (arXiv 2511.14406)

Chain-of-Bit-Flips: fault-based backdoors in FL

B. Vuillod & K. Hector



36 → 165

bit-flips: offline list → chained faults

≤ 10

bit-flips per participation

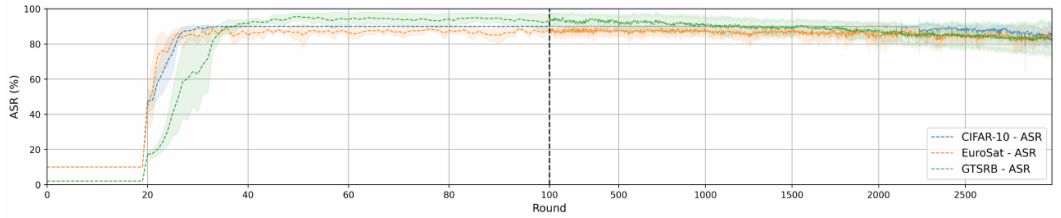
19

participations of the targeted client

94 %

ASR on ResNet-18

- **Offline:** from the public pre-trained model, pick the weights to fault and optimize a trigger (WBP, centralized)
- **Obstacle:** FedAvg dilutes a local fault among the |P| participants
- **CoBF:** estimate the other clients' contributions and schedule chains of intermediate flips that steer each weight to its target
- **Rowhammer viability:** one flip per weight, one weight per DRAM page



ResNet-18: backdoor still active > 1000 rounds after the attack (VGG-16: > 60 % ASR)

Mitigations: keep a high learning rate; norm clipping, since bypassing it multiplies the fault budget

Vuillod*, Hector*, Moëllic, Dutertre, Potin, Model poisoning against federated model adaptation with chain of bit-flips, ACNS/AIHWS 2026 (arXiv 2606.09548)

Conclusion

Secure the chain

Alignment, LoRA, federated aggregation, quantization and memory can each create or reactivate a threat.

Budget-aware threat models

Count faults per shot, per DRAM page, per round. Number formats change the bit search space and per-bit sensitivity.

Evaluation rigor

Three traps giving a false sense of security: high BFA variance, attack windows too short in FL, naive jailbreak metrics.

Perspectives

Defenses that survive compression; bit-flips on quantized LLMs (integrity & availability); alignment-aware deployment.

Supported by PEPR Cybersécurité COMPROMIS (France 2030)
IRT Nanoelec (ANR-10-AIRT-05)
MIAI@Grenoble Alpes (ANR-19-P3IA-0003)
ANR AI.MMUNITY
computing: GENCI Jean Zay